OIPE JC25 AUG 16 2004 PATENT & TRADEMARK OFFICE

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.	: 10/767,454	Confirmation No.:	3942
Applicant	: Richard C.BEESLEY, et al.		
Filed	: January 30, 2004		
TC/A.U.	: 2155		
Examiner	: To Be Assigned		
Docket No.	: 038819.53225US		
Customer No.	: 23911		
Title	: Secure Network Browsing		

CLAIM OF PRIORITY UNDER 35 U.S.C. § 119

Mail Stop Missing Parts
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

The benefit of the filing date of prior foreign application No. 0302263.9, filed in United Kingdom on January 31, 2003, is hereby requested and the right of priority under 35 U.S.C. § 119 is hereby claimed.

In support of this claim, filed herewith is a certified copy of the original foreign application.

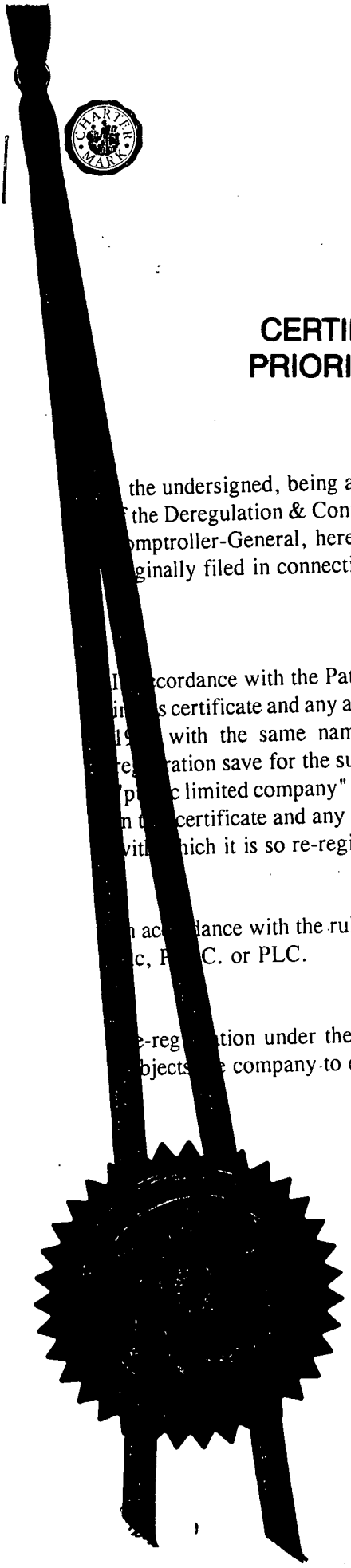
Respectfully submitted,

August 16, 2004

Gary R. Edwards
Registration No. 31,824

CROWELL & MORING LLP
Intellectual Property Group
P.O. Box 14300
Washington, DC 20044-4300
Telephone No.: (202) 624-2500
Facsimile No.: (202) 628-8844

GRE:ms #333214



The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

INVESTOR IN PEOPLE

CERTIFIED COPY OF PRIORITY DOCUMENT

the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the omptroller-General, hereby certify that annexed hereto is a true copy of the documents as ginally filed in connection with the patent application identified therein.

cordance with the Patents (Companies Re-registration) Rules 1982, if a company named s certificate and any accompanying documents has re-registered under the Companies Act with the same name as that with which it was registered immediately before re- ration save for the substitution as, or inclusion as, the last part of the name of the words c limited company" or their equivalents in Welsh, references to the name of the company certificate and any accompanying documents shall be treated as references to the name hich it is so re-registered.

lance with the rules, the words "public limited company" may be replaced by p.l.c., C. or PLC.

tion under the Companies Act does not constitute a new legal entity but merely company to certain additional company law rules.

Signed JEvans.

Dated 13 February 2004

Pate Form 1/77

Patents Act 1977
(Rule 16)

The Patent Office

1/77

Request for grant of a patent

(See the notes on the back of this form. You can also get an explanatory leaflet from the Patent Office to help you fill in this form)

The Patent Office

Cardiff Road
Newport
Gwent NP10 8QQ

1. Your reference — 2003P01280 GB / R76 / NC / RAN

2. Patent application number *(The Patent Office will fill in this part)* — 0302263.9

3. Full name, address and postcode of the or of each applicant *(underline all surnames)*

ROKE MANOR RESEARCH LIMITED
Old Salisbury Lane, Romsey
Hampshire SO51 0ZN

Patents ADP number *(if you know it)* — 56154550006

If the applicant is a corporate body, give the country/state of its incorporation — UNITED KINGDOM

4. Title of the invention — SECURE NETWORK BROWSING

5. Name of your agent *(If you have one)* — Neil Condon

"Address for service" in the United Kingdom to which all correspondence should be sent *(including the postcode)*

Siemens Shared Services
Intellectual Property Department
Siemens House, Oldbury
Bracknell, Berkshire, RG12 8FZ

Patents ADP number *(if you know it)* — 85827690001

Patents Form 1/77

IDNR: 7100 / V 02-1.00 / B: Val

Pat Form 1/77

Patents Act 1977
(Rule 16)

The Patent Office

1/77

		Country	Priority application number *(if you know it)*	Date of filing *(day / month / year)*
6.	If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each these earlier applications and *(if you know it)* the or each application number			
7.	If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application			
8.	Is a statement of inventorship and of right to grant of a patent required in support of this request? (Answer 'Yes' if: *a) any application named in part 3 is not an inventor, or* *b) there is an inventor who is not named as an applicant, or* *c) any named applicant is a corporate body.* *See note (d)*		Yes	

9. Enter the number of sheets for any of the following items you are filing with this form. Do not count copies of the same document

Continuation sheets of this form	0
Description	4
Claim(s)	0
Abstract	0
Drawing(s)	1 only

10. If you are also filing any of the following, state how many against each item.

Priority documents	0
Translation of priority documents	0
Statement of inventorship and right to grant a patent *(Patents Form 7/77)*	0
Request for preliminary examination and search *(Patents Form 9/77)*	0
Request for substantive examination *(Patents Form 10/77)*	0
Any other documents *(please specify)*	

Pat ; Form 1/77

Patents Act 1977
(Rule 16)

The Patent Office

1/77

11.	I/We request the grant of a patent on the basis of this application	
	Signature Neil Condon Intellectual Property Department	Date 31.01.2003
12. Name and daytime telephone number of Person to contact in the United Kingdom	Neil Condon	+ 44 1344 85 0066

DUPLICATE)

SECURE NETWORK BROWSING

Web browsers present a local security and privacy risk on a PC

- URLs, passwords, cookies, cached web pages and other information is stored on the PC's hard disk
- Not possible to turn off every data gathering option
  - Internet Explorer insists on at least a 1MB web page cache
  - Extremely difficult to permanently delete everything
  - Magnetic 'impression' left on the hard disk
  - Windows/Internet Explorer locks some files preventing deletion at all!

Consider accessing personal information (online banking, web mail etc) at:

- Internet cafes
- Hot desking/shared PCs
- Libraries
- Universities
- Hotel/Conference business facilities

Lack of administrator privileges compounds the problem of securely configuring the browser and 'tidying up' after yourself or installing new, more secure software.

Data is left on a public PC that could be used to steal money or private information from the original user.

Consider the following example scenario:

1) User A enters an Internet Cafe and is allocated a PC to use.
2) User A wants to buy a book from Amazon.com so the user types http://www.amazon.com into the web browser.
3) The web browser retrieves the web page. The URL (i.e. http://www.amazon.com) is stored to hard disk as is the web page.
4) User A types in their username and password to access their account. The web browser stores this information.
5) User A chooses a book and goes to the check out page. They type in their credit card details and click finish. The credit card details are also stored by the web browser.
6) User A finishes their web browsing and leaves the Internet Cafe.
7) User B enters the Internet Cafe and is allocated the same PC.
8) Browsing the harddisk they find all of the information stored by User A.
   User B logs onto Amazon as User A, changes the home delivery address and uses the same credit card details to order 100 books.

There are products available (for Internet Explorer only) that can be installed on a PC to delete files after Internet Explorer has created them. However, this requires the ability to install new software and a reboot to remove all files. This is simply not possible on a public PC to install new software.

Alternatively, the browser manufactures may choose to increase the functionality of their web browsers to address this issue. However, this still requires the new web browser to be installed and correctly configured on the public PC. The proposed solution removes any reliance on any supposed client side secure configuration by the system administrator.

A variation on this is a method whereby the web server can negotiate with the web browser for secure local environment. However this still has a flaw in requiring trust of the validity and configuration of the locally installed web browser.

Typically Internet secure solutions are concerned with the link between the client and the server. This solution addresses security on the client itself.

A user may not be able to download new application software on a public terminal. Therefore any solution must be operable within the confines of software already installed on that terminal i.e. a web browser with a Java environment (Java/browser installation penetration is in the 99.9+ range).

In an embodiment of the present invention, the user browses to a web page and downloads a new Java applet. This applet is in itself a new web browser. However, it has been written from a secure aspect with no physical disk storage required, no logging of data or other traces left. The user then can interact with this browser within a browser with the knowledge no residue is left on the public terminal.

For example:

1) User A enters an Internet Cafe and is allocated a PC to use.
2) User A visit a web site where they can download a Java web browser applet. For example, they type http://www.javabrowser.com into the web browser (e.g. Internet Explorer). This URL is stored by the web browser.
3) Internet Explorer downloads the Java web browser applet and runs it, displaying the browser applet within the main Internet Explorer window. This applet is stored to hard disk by Internet Explorer.
4) User A wants to buy a book from Amazon.com so the user types http://www.amazon.com into the Java web browser.
5) The Java web browser retrieves the web page. The URL (i.e. http://www.amazon.com) is not stored.
6) User A types in their username and password to access their account. This information is not stored
7) User A chooses a book and goes to the check out page. They type in their credit card details and click finish. The credit card details are not stored.
8) User A finishes their web browsing and leaves the Internet Cafe.
9) User B enters the Internet Cafe and is allocated the same PC.
10) Browsing the harddisk they find all of the information stored by User A.

User B finds user A visited http://www.javabrowser.com and downloaded a Java web browser applet. This is the only information they can find.

The ability to download and use a trusted web browser in an untrusted environment and preserve the security of the downloaded browser is advantageous.

Conceptually, this screen mock up illustrated in figure 1 shows what a Java web browser within the standard Internet Explorer could look like:

Due to the security environment of Java applets they can only communicate back to web server they were download from. Therefore to enable Internet wide access the web server must also be a web proxy server. Alternatively, the browser may be downloaded from a single site such as an Internet bank and only communication with that bank's web site possible. To prevent the need to download the browser multiple times every time a new site is visited there could be a core web browser that is download from one site and Internet enabling plugins for other sites. The use of a proxy will also allow other traffic than web to be accessible from within the main browser.

To minimise download time in general this modular approach could also be used to only download the web browser components as they are required. For example, only download a plug in to render a .GIF format image when one is encountered.

The applet must follow good practice for security software. Volatile memory can still be interrogated. Therefore the applet should not store data longer than necessary and destroy it by overwriting it rather than simply returning it to the system pool. If necessary all other data could be stored in encrypted format in RAM. As part of the download procedure or to verify the integrity of a previously installed web browser applet still on the public terminal there can also be a mechanism of performing a check using a standard method such as a CRC or hash test.

Software exists to monitor key strokes. If this has been installed on a public terminal a user's password or other details could be found. However, now that there is secure control over the web browser environment extra functionality such as a pop up mouse driven keyboard can be included. It may be possible to monitor the movement of the mouse so the keyboard layout can be randomised to prevent playing back of the mouse movements at a later date to find the keys clicked on.

Following on from this, the downloaded web browser may be modified to match particularly requirements such as from an Internet bank whilst still using Internet and web technologies. This would be a variation on the original idea but still use the security and privacy concepts.

In an alternative embodiment, a server side web browser is run with the display echoed to the local terminal. The terminal software will still require many of the features of a web browser such as supporting pull down menus, secure socket layers etc so this is simply abstracting where some of the functionality lies but still preserving the concept of not storing information locally. In detail, the user downloads a Java applet. This Java applet is similar to the web browser applet in that it is securely written to not require access to the hard disk and to not cache information. This Java applet communicates with a web browser-like process running on a server. Each key press or mouse movement is send to this web browser process. The process interprets these actions within the context of a web browser. For example, if the user types in http://www.bbc.co.uk in the Java applet, this text is sent to the server, the server will input the text to it's web browser process and retrieve the web page. The web page is then sent in a graphical format (i.e. not HTML) to the Java applet which displays it. As the graphical image is sent to the Java applet and not the main web browser (i.e. Internet Explorer) no caching of the image will occur. Furthermore, as the URL http://www.bbc.co.uk was typed in the Java applet and not Internet Explorer it also will not be cached. For example:

1) User A enters an Internet Cafe and is allocated a PC to use.
2) User A visit a web site where they can download a Java applet. For example, they type http://www.javaapplet.com into the web browser (e.g. Internet Explorer). This URL is stored by the web browser.
3) Internet Explorer downloads the Java applet and runs it, displaying the applet within the main Internet Explorer window. This applet is stored to hard disk by Internet Explorer.
4) User A wants to buy a book from Amazon.com so the user types http://www.amazon.com into the Java applet.
5) The Java applet sends these key presses to the Java browser server
6) The Java browser server retrieves the web page, formats it graphically and sends it to the Java applet.
7) The Java applet displays the graphic. This graphic is not stored to hard disk.
8) User A types in their username and password to access their account. This information is sent via the Java browser server. The username and password is not stored.
9) User A chooses a book and goes to the check out page. They type in their credit card details and click finish. The credit details details and mouse click are sent to the Java browser server. The credit card details are not stored.
10) User A finishes their web browsing and leaves the Internet Cafe.
11) User B enters the Internet Cafe and is allocated the same PC.
12) Browsing the harddisk they find all of the information stored by User A.
13) User B finds user A visited http://www.javaapplet.com and downloaded a Java applet. This is the only information they can find.

Thus embodiments of the invention allow a user to access the Internet from a public terminal in a secure and private manner to allow the user to view sensitive information such as bank details or email. The method is not dependant on the security and privacy of the public terminal itself thus providing a self contained security and privacy solution.

The above examples make use of a Java applet. This is exemplary only and it will be appreciated that other downloadable applications may be configured to perform the function of the applets described above.

CLAIMS

1. A method of enhancing data security at a terminal connected to a public data network, the method comprising; transmitting over the network from the terminal to a server a request for a network communications programme stored on the server to be downloaded to the terminal, receiving the communications programme at the terminal and running the communications programme at the terminal, wherein the communications programme is configured so that data received as user input at the terminal by the programme for transmission into the network is transmitted into the network without a record of the data being stored at the terminal or that user requested data received at the terminal by the programme from the network is presented to the user without a record of the data being stored at the terminal.

BEST AVAILABLE COPY

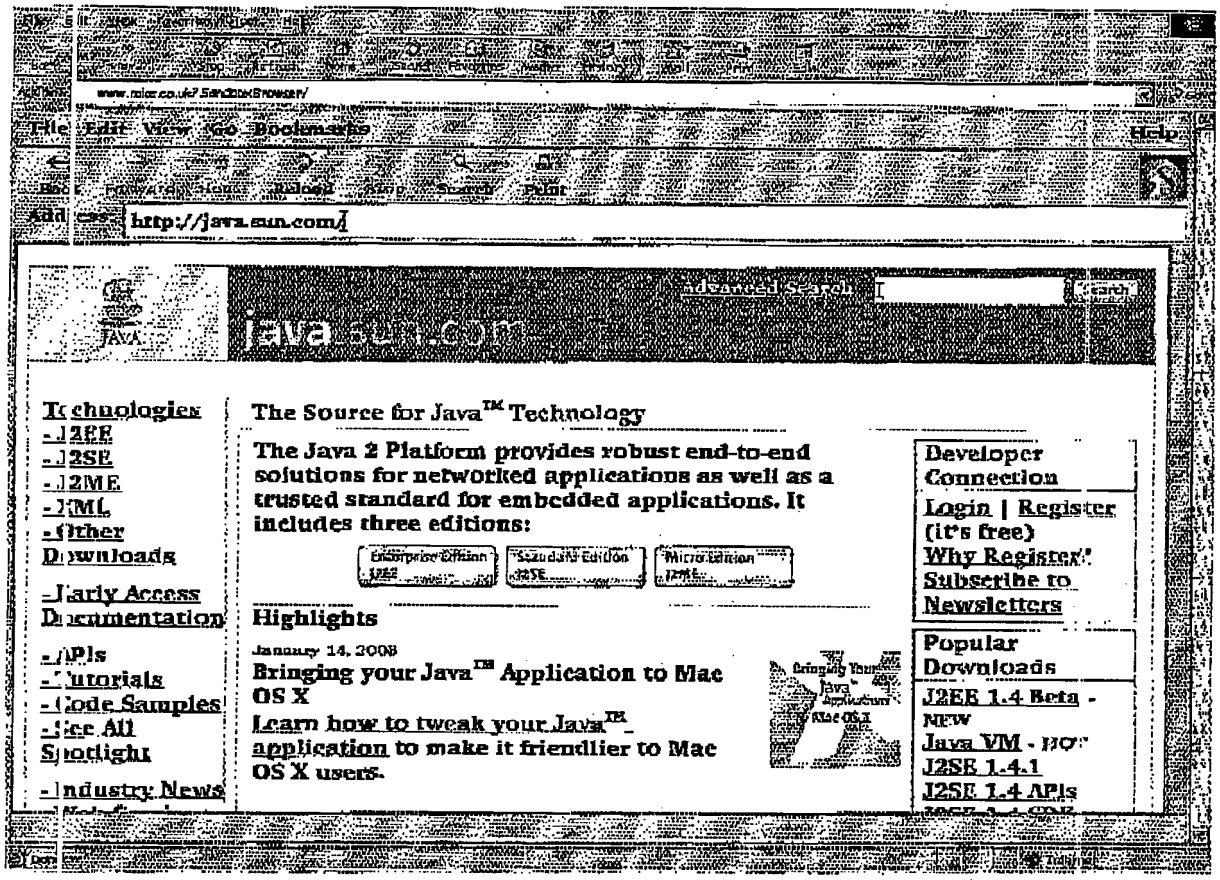


FIGURE 1